

# Stopping Criteria for Iterative Solution Methods

Lecture notes for ME 448/548

Gerald Recktenwald\*

February 14, 2012

## 1 Direct and Iterative Methods

The goal is to solve

$$Ax = b \tag{1}$$

where  $A$  is an  $n \times n$  matrix,  $x$  and  $b$  are  $n$ -element column vectors. A direct method obtains the solution in a finite number of steps. In other words, for a given system we know, *a priori*, exactly how many mathematical operations are required to solve  $Ax = b$  by a direct method. Gaussian elimination with backward substitution is a *direct method*.

The formal solution to Equation (1) is

$$x = A^{-1}b \tag{2}$$

It is a *bad idea* to compute  $x$  from Equation (2). Instead, we perform a procedure, for example Gaussian elimination with backward substitution, that has the same effect as the multiplication  $A^{-1}b$  but without all the round-off and computational efficiency of a literal implementation of Equation (2)

*Iterative Methods* obtain a sequence of approximations to the solution

$$x^k \quad k = 0, 1, 2, \dots$$

such that  $(b - Ax^k) \rightarrow 0$  as  $k \rightarrow \infty$ . In practice the number of iterations of an iterative method is finite. Thus, we don't know how many mathematical operations are required to obtain a solution. At some point we merely declare that the solution is *close enough*. What is a rational criterion for deciding when to stop?

---

\*Mechanical and Materials Engineering Department, Portland State University, Portland, OR, 97201, [gerry@me.pdx.edu](mailto:gerry@me.pdx.edu)

## 2 Residual

The residual

$$r = b - Ax$$

is zero when  $x$  is the solution to  $Ax = b$ . For an iterative method the residual at iteration  $k$  is

$$r^k = b - Ax^k \quad k = 0, 1, 2, \dots \quad (3)$$

If an iterative methods *converges* then  $r^k \rightarrow 0$  as  $k \rightarrow \infty$ .

## 3 Convergence

Let  $x^*$  designate the exact solution. If the exact solution exists (i.e., if  $A$  is nonsingular), then  $x^*$  is obtained only after an infinite number of iterations of an iterative method.

Substitute  $Ax^*$  for  $b$  into Equation (3)

$$r^k = Ax^* - Ax^k = A(x^* - x^k) \quad \implies \quad x^* - x^k = A^{-1}r^k.$$

Therefore, the size of the error at the  $k^{th}$  iteration is

$$\|x^* - x^k\| = \|A^{-1}r^k\|.$$

From the definition of matrix norms, (see, e.g., [2, p. 42])

$$\|A^{-1}r^k\| \leq \|A^{-1}\| \|r^k\|$$

so

$$\|x^* - x^k\| \leq \|A^{-1}\| \|r^k\|.$$

Dividing through by the scalar,  $\|x^*\|$ , gives

$$\frac{\|x^* - x^k\|}{\|x^*\|} \leq \|A^{-1}\| \frac{\|r^k\|}{\|x^*\|}. \quad (4)$$

To continue we need a replacement for  $\frac{1}{\|x^*\|}$  on the right hand side of Equation (4). Consider

$$\|b\| = \|Ax^*\| \leq \|A\| \|x^*\| \quad \implies \quad \frac{1}{\|x^*\|} \leq \frac{\|A\|}{\|b\|} \quad (5)$$

Substituting the right hand side of Equation (5) into the right hand side of Equation (4) gives

$$\frac{\|x^* - x^k\|}{\|x^*\|} \leq \|A^{-1}\| \|r^k\| \frac{\|A\|}{\|b\|}. \quad (6)$$

Introduce the condition number of  $A$

$$\kappa(A) \equiv \|A\| \|A^{-1}\|$$

into Equation (6) to get

$$\boxed{\frac{\|x^* - x^k\|}{\|x^*\|} \leq \kappa(A) \frac{\|r^k\|}{\|b\|}}. \quad (7)$$

Therefore, the relative error in the solution to  $Ax = b$  incurred by stopping at iteration  $k$  can be estimated by the size of  $\|r^k\|/\|b\|$ . If  $A$  is ill-conditioned, i.e., if  $\kappa(A)$  is large, then  $\|r^k\|/\|b\|$  is not a reliable indicator of convergence.

## 4 Implementation

The preceding analysis suggests that a stopping criterion is  $\|r\| < \epsilon\|b\|$ , where  $\epsilon$  is a tolerance. Thus, the iterations continue until

$$\frac{\|r^k\|}{\|b\|} < \epsilon \quad (8)$$

$\epsilon$  is a small value, but much larger than machine precision. Typically  $5 \times 10^{-3} > \epsilon > 1 \times 10^{-6}$ , but the value of  $\epsilon$  will depend on the problem being solved and the cost of the iterations.

The following MATLAB code snippet shows how a convergence test might be implemented. Suppose that there is a routine called `itersolver` that performs one iteration to update  $x$  from the fixed  $A$  and  $b$  and the current guess at  $x$ .

```
A = ...           % A and b are defined by the problem to be solved
b = ...
x = ...           % initial guess
tol = ...         % Convergence tolerance, e.g. tol = 5.0e-4
normb = norm(b); % compute ||b|| only once
itermax = ...    % Limit on the number of iterations

while i<itermax
    r = b - A*x;           % Compute residual with previous x
    normr = norm(r)       % Save for printing
    if normr/normb < tol, break; end % Test before solving; exit loop if true
    x = itersolver(A,b,x); % Update the solution
    i = i + 1;
    fprintf(' %4d %12.3e\n',i,normr);
end
```

Note that the convergence tolerance is `tol`, not `eps` because `eps` is a built-in variable equal to machine precision,  $\text{eps} \sim 2 \times 10^{-16}$ .

## 5 Alternative Criterion

In CFD codes a typical stopping criterion is

$$\frac{\|r^k\|}{\|r^0\|} < \epsilon \quad (9)$$

where  $r^0 = b - Ax^0$  is the residual based on the initial guess,  $x^0$ . If  $x^0 = 0$ , i.e., if the initial guess is a vector of zeros, then  $r^0 = b$ . However, for any other  $x^0$ , the value of  $r^0$  may be large and unrelated to the true solution. If  $r^0$  is artificially large then the iterations may be stopped prematurely.

Barret et. al [1, Chapter 4] consider Equation (9) to be a *Dubious Criterion* and recommend against using it. However, it *is* widely used.

## 6 Nonlinear Systems of Equations

A nonlinear system of equations can be written in the familiar form

$$Ax = b, \tag{10}$$

but now  $A = A(x)$  and  $b = b(x)$ . Solution methods for nonlinear systems are generalizations of the root-finding methods for solving scalar equations of the form  $f(\xi) = 0$ . When a system of nonlinear equations is to be solved, the  $\xi$  becomes a vector of unknowns, and solving  $Ax = b$  is equivalent to finding the  $x$  that gives

$$f(x) = Ax - b = 0, \tag{11}$$

where  $f(x)$  is a *vector valued* function of  $x$ . Alternatively, one can express the nonlinear problem in terms of the residual

$$r = b - Ax = -f(x). \tag{12}$$

An iterative method to solve Equation (10) can be written as

$$x^{k+1} = x^k + \Delta x^k, \quad k = 1, 2, \dots, \tag{13}$$

Remember that  $x^k$  is *not* the  $k$ th power of  $x$ . Using this equation requires a procedure for computing the update vector  $\Delta x^k$  from the *linearized* coefficient matrix and right-hand-side vector

$$A^k = A(x^k), \quad b^k = b(x^k).$$

Before the update to  $x^k$  is computed, the vector  $f^k = f(x^k)$ , or

$$f^k = A^k x^k - b^k, \tag{14}$$

will not be zero unless  $x^k$  is the solution to the nonlinear problem. Thus, convergence of the iterative algorithm is monitored by checking  $\|f^k\|$ , which, because  $\|y\|$  is always positive for any  $y$ , is equal to  $\|r^k\|$ . The logic of the iterative solution to Equation (10) is contained in Algorithm 1

### Algorithm 1 Iterative Solution of Nonlinear Systems

```
initialize:  $x = x^{(0)}$ 
for  $k = 0, 1, 2, \dots$ 
   $A^k = A(x^k)$            Linearize  $A$ 
   $b^k = b(x^k)$            and  $b$ 
   $f^k = A^k x^k - b^k$ 
  if  $\|f^k\|$  is small enough, stop
   $\Delta x^{k+1} = \dots$        Compute the update
   $x^{k+1} = x^k + \Delta x^k$ 
end
```

Two very important observations:

1. The non-linear system *must use* iterations to approach the solution. There is no direct method to solve  $Ax = b$  when  $A = A(x)$  and/or  $b = b(x)$ .
2. The update step,  $\Delta x^{k+1}$  requires work equivalent to solving  $Ax = b$ . In other words, each step of the iterative solver requires solution of a linear(ized) system of equations.

## 6.1 Inner and Outer Iterations

Iterative methods are often used to compute the update  $\Delta x^{k+1}$  from the current estimates of the linearized coefficients  $A^k$  and right hand side  $b^k$ . This leads to iterations within iterations.

**Outer Iterations:** Each step of the updating the non-linear system of equations  $A^k x^{k+1} = b^k$  is called an *outer iteration*.

**Inner Iterations:** Each step of the iterative solver applied to the frozen coefficients  $A^k$  and  $b^k$  for finding  $x^{k+1}$  is called an *inner iteration*.

We need two different iteration counters. In §1 through §5,  $k$  was the counter for the inner iterations. In practice, this is not a problem because the iterative solver has internal variables that are not shared with the rest of the code.

Note that we only really care about the residual of the outer iterations. We don't want to spend too much effort on the inner iterations because during the next outer iterations the  $A^{k+1}$  and  $b^{k+1}$  are different than  $A^k$  and  $b^k$ . In practice, the inner iterations need to be repeated enough that the update  $\Delta x^{k+1}$  moves the non-linear solution in the right direction.

## References

- [1] R. Barret, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. van der Vorst. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*. SIAM, Philadelphia, 1994.
- [2] G. Golub and J. M. Ortega. *Scientific Computing: An Introduction with Parallel Computing*. Academic Press, Inc., Boston, 1993.